



## Beschreibung

## Verfahren zur Umsetzung von Schnittstellendefinitionen und Zwischenformattabelle dafür

5

Die vorliegende Erfindung betrifft ein Verfahren zur Umsetzung von Schnittstellendefinitionen innerhalb von Quellprogrammcode in ein Zwischenformat und aus einer Zwischenformattabelle in Zielprogrammcode sowie eine dafür geeignete Zwischenformattabelle.

In Programmen werden sogenannte Schnittstellen zum Austausch von Daten zwischen einzelnen Programmmodulen beziehungsweise Programmobjekten verwendet. Besonders intensiven Gebrauch von Schnittstellendefinitionen machen naturgemäß Modellierer/Programmierer von Hardware-Beschreibungen. Hierzu werden spezielle Hardware-Beschreibungssprachen verwendet, die das Verhalten der einzelnen simulierten Komponenten sowie die elektrische Verknüpfung der Komponenten miteinander beschreiben. Im Bereich der Hardware-Beschreibungssprachen ist es üblich, den Programmcode in einen Bereich zu teilen, der der Definition der Architekturen der Bauteile dient und in einen zweiten Bereich, der ausschließlich die Schnittstellen zwischen den Bauteilen definiert.

25

Um einen Datenaustausch zwischen verschiedenen Hardware-Entwicklungsprojekten zu ermöglichen, ist es notwendig, Programmcode von einer Ausgangshardware-Beschreibungssprache in eine Zielbeschreibungssprache zu übersetzen, so daß diese von bei einem anderen Projekt verwendeten Hardware-Simulatoren beziehungsweise Hardware-Syntheseprogrammen bearbeitet werden kann. Auch bei reinen Softwareprojekten kann eine Übersetzung zwischen einer Programmiersprache und einer anderen notwendig sein, beispielsweise um sie auf anderen Hardware-Plattformen lauffähig zu machen. Typische Beispiele für Hardware-Beschreibungssprachen, zwischen denen eine Konvertierung erfolgen kann, sind VHDL und VERILOG. Es gibt jedoch auch wei-

35

tere Programmiersprachen im Hardware-Beschreibungsbereich,  
zwischen denen eine Konvertierung notwendig sein kann.

Die unmittelbare Umsetzung zwischen Programmiersprachen ist  
5 komplex und fehleranfällig. Es ist daher möglich, daß eine  
Zielschnittstellendefinition nicht mehr mit der Ausgangs-  
schnittstellendefinition übereinstimmt, so daß es zu Fehlern  
bei der Hardwaresimulation kommt. Auf Grund der Komplexität  
10 sowohl des Ausgangsprogrammcodes als auch des Zielpro-  
grammcodes ist eine Verifizierung der Korrektheit der Umset-  
zung durch einen Fachmann eine ebenfalls schwierige und feh-  
lerbehaftete Aufgabe. Zudem wird für jedes mögliche Sprachen-  
paar von Programmiersprachen, zwischen denen eine Umsetzung  
15 erfolgen soll, ein Übersetzungsprogramm benötigt. Bei der  
Vielzahl von Programmiersprachen, bei denen Schnittstellende-  
finitionen eine Rolle spielen, resultiert daraus eine große  
Zahl notwendiger Umsetzer hohen Komplexitätsgrads.

Der vorliegenden Erfindung liegt daher die Aufgabe zugrunde,  
20 ein Übersetzungssystem bereitzustellen, das universeller ein-  
setzbar ist, von seinem Aufbau her weniger komplex, und das  
verifizierbare Ergebnisse produziert. Erfindungsgemäß wird  
diese Aufgabe gelöst durch das Verfahren zur Umsetzung von  
Schnittstellendefinitionen innerhalb von Quellprogrammcode in  
25 ein Zwischenformat gemäß dem unabhängigen Patentanspruch 1,  
ein Verfahren zur Umsetzung von Schnittstellendefinitionen  
aus einer Zwischenformattabelle in Zielprogrammcode gemäß dem  
unabhängigen Patentanspruch 17, sowie eine dafür geeignete  
Zwischenformattabelle gemäß dem unabhängigen Patentanspruch  
30 25.

Weitere vorteilhafte Ausgestaltungen, Aspekte und Details der  
Erfindung ergeben sich aus den abhängigen Patentansprüchen,  
der Beschreibung und den beigefügten Zeichnungen.

35 In einem Aspekt ist die Erfindung gerichtet auf die Umsetzung  
zu und von einem Zwischenformat in einer tabellarischen Form.

Die Umsetzung erfolgt dabei durch ein Programm in einer Datenverarbeitungsanlage.

5 In einem weiteren Aspekt ist die Erfindung gerichtet auf ein Zwischenformat, das als Bindeglied zwischen verschiedenen Umsetzungssystemen fungiert.

10 In noch einem Aspekt ist die Erfindung gerichtet auf Umsetzungssysteme, deren Zwischenergebnisse trotz einer Übersetzungsautomatik manuell inspizierbar und änderbar sind.

15 Die erfindungsgemäßen Verfahren werden in einer Rechenanlage durchgeführt und die Zwischenformattabelle in einer Rechenanlage gespeichert, auch wenn diese ausgedruckt bzw. auf einem Datensichtgerät dargestellt werden kann. Unter einer Rechenanlage im Sinne der vorliegenden Erfindung ist dabei jede Einrichtung zu verstehen, die die erfindungsgemäßen Verfahren ohne menschliche Eingriffe durchführen können (außer zum Starten der Verfahren) und die Zwischenformattabelle selbständig generieren können. Geeignete Rechenanlagen sind zum Beispiel Personal-Computer, Workstations, Großrechner, Taschenrechner etc.

25 Unter einem Programmcode ist im Sinne der vorliegenden Erfindung eine Abfolge von Anweisungen in einer bestimmten Programmiersprache zu verstehen, die in einem funktionellen Zusammenhang stehen. Ein Quellprogrammcode im Sinne der vorliegenden Erfindung ist ein solcher, von dem für eine Umsetzung in einen anderen Programmcode ausgegangen wird, der dann als Zielprogrammcode bezeichnet wird.

35 Ein Objekt des Quellprogrammcodes ist im Sinne der vorliegenden Erfindung eine im Programmcode enthaltene zusammengehörige Abfolge von Anweisungen in einer Programmiersprache. Bezogen auf Hardware-Programmiersprachen kann ein Objekt gleichgesetzt werden mit der Definition einer Hardwarekomponente.

Bezogen auf solche Programmiersprachen, bei denen Schnittstellendefinitionen in einem eigenständigen Teil des Programmcodes erscheinen, kann ein Objekt auch verstanden werden als die Summe der Schnittstellendefinitionen einer Komponente  
5 der zu simulierenden Hardware.

Unter einer Schnittstelle im Sinne der vorliegenden Erfindung ist ein Ein- oder Ausgang zu verstehen, über den Daten zwischen Objekten hin- und/oder hergeleitet werden, beispielsweise  
10 weise über globale Variablen oder in einer anderen Weise.

Eine interne Schnittstelle im Sinne der vorliegenden Erfindung ist eine solche, die eine Verbindung zwischen zwei Objekten innerhalb eines Programmcodes vornimmt.  
15

Eine externe Schnittstelle im Sinne der vorliegenden Erfindung dient der Herstellung einer Verbindung von einem Objekt innerhalb des Programmcodes mit außerhalb des Programmcodes liegenden Schnittstellen anderer Objekte, beispielsweise wenn  
20 auf Grund des modularen Aufbaus einer Hardwaresimulation verschiedene Teile von Programmcode verwendet werden, die miteinander über solche externen Schnittstellen verbunden sind.

Schnittstellen können Eingangsschnittstellen oder Ausgangsschnittstellen sein, je nachdem in welche Richtung die Daten fließen, wobei es ebenfalls möglich ist, daß eine Schnittstelle gleichzeitig beide Funktionen einer Eingangsschnittstelle und einer Ausgangsschnittstelle übernimmt. In bestimmten Beschreibungssprachen ist es auch möglich, Schnittstellen zu  
25 definieren, deren Informationsflußrichtung nicht festgelegt, das heißt unbestimmt ist.  
30

Eine interne Verknüpfung im Sinne der vorliegenden Erfindung ist demgemäß die zwischen zwei internen Schnittstellen erfolgte Verbindung. Diese kann bei Hardware-Simulationen auch  
35 als Signal bezeichnet werden und simuliert in der Tat dann eine elektrische Leitung zwischen zwei Hardwarekomponenten.

Eine externe Verknüpfung ist entsprechend einer Verknüpfung einer externen Schnittstelle mit einer Schnittstelle, die außerhalb des betrachteten Programmcodes liegt.

5

Von einer Verknüpfung wird hier in Zusammenhängen gesprochen, in denen eine interne Verknüpfung und/oder eine externe Verknüpfung vorkommen kann.

- 10 Die Erfindung ist zunächst gerichtet auf ein Verfahren zur Umsetzung von Schnittstellendefinitionen innerhalb von Quellprogrammcode in ein Zwischenformat mittels einer das Verfahren ausführenden Rechenanlage mit folgenden Schritten:

- 15 A: Identifizieren von zumindest einem Objekt im Quellprogrammcode;

B: Identifizieren von zumindest einer Schnittstelle bei dem zumindest einen identifizierten Objekt;

20

wobei zumindest eine der identifizierten Schnittstellen eine interne Schnittstelle zur Herstellung einer Verknüpfung von Objekten innerhalb des Quellprogrammcodes und/oder zumindest eine der identifizierten Schnittstellen eine externe Schnittstelle zur Herstellung einer Verknüpfung von einem Objekt mit außerhalb des Quellprogrammcodes liegenden Schnittstellen sein kann; und

25

- 30 wobei die zumindest eine Schnittstelle eine Eingangs- und/oder eine Ausgangsschnittstelle sein kann;

- C: Identifizieren von zumindest einer internen Verknüpfung zwischen zumindest einer Ausgangsschnittstelle und zumindest einer Eingangsschnittstelle zwischen zumindest zwei Objekten
- 35 und/oder

Identifizieren von, zumindest einer externen Verknüpfung der  
zumindest einen externen Schnittstelle;

- 5 D:Erstellen einer zumindest zweidimensionalen Zwischenfor-  
mattabelle mit in einer ersten Dimension angeordneten Reihen,  
in einer zweiten Dimension angeordneten Reihen und Zellen an  
den Kreuzungspunkten der ersten und zweiten Reihen,

- 10 wobei Reihen der ersten Dimension Bezeichnungen für jedes des  
zumindest einen identifizierten Objekts zugewiesen werden;

- 15 wobei Reihen der zweiten Dimension Bezeichnungen für jede der  
zumindest einen identifizierten Verknüpfungen zugewiesen wer-  
den; und

- 20 wobei jeweils in diejenigen Zellen, die am Kreuzungspunkt ei-  
ner der Reihen der ersten Dimension mit der Bezeichnung eines  
identifizierten Objekts und einer der Reihen der zweiten Di-  
mension mit der Bezeichnung einer identifizierten internen  
Verknüpfung liegen, Bezeichnungen für die zu sowohl dem je-  
weiligen identifizierten Objekt als auch der identifizierten  
internen Verknüpfung gehörigen Ausgangsschnittstelle und/oder  
Eingangsschnittstelle eingesetzt werden; und/oder

- 25 wobei jeweils in diejenigen Zellen, die am Kreuzungspunkt ei-  
ner der Reihen der ersten Dimension mit der Bezeichnung eines  
identifizierten Objekts und einer der Reihen der zweiten Di-  
mension mit der Bezeichnung einer identifizierten externen  
Verknüpfung liegen, Bezeichnungen für die zu sowohl dem je-  
30 weiligen identifizierten Objekt als auch der identifizierten  
externen Verknüpfung gehörigen Ausgangsschnittstelle und/oder  
Eingangsschnittstelle eingesetzt werden.

- 35 Unter einer Zuweisung im Sinne der vorliegenden Erfindung ist  
hierbei zu verstehen, daß eine Reihe dafür vorgesehen ist,  
Informationen über die ihr zugewiesenen Eigenschaften, bei-  
spielsweise eines Objekts oder einer Verknüpfung, zu enthal-

ten, ohne daß notwendigerweise eine der Zellen der Reihe eine entsprechende Bezeichnung oder Kennzeichnung trägt.

- 5 Unter Einsetzen im Sinne der vorliegenden Erfindung wird verstanden, daß in der Tat eine Angabe, welcher Art auch immer, in die Zelle einer Reihe oder in die Kreuzungszelle von zwei Reihen hineingeschrieben werden, so daß die entsprechende Zelle einen Inhalt hat.
- 10 Eine Schnittstelle kann hierbei auch gleichzeitig die Funktion einer Eingangs- und Ausgangsschnittstelle haben, oder eine Schnittstelle unbestimmter Datenflußrichtung sein.
- 15 Jede Verknüpfung erhält hierbei eine Reihe der zweiten Dimension zugewiesen, während jedes der identifizierten Objekte im Quellprogrammcode eine Reihe der ersten Dimension zugewiesen bekommt, die durch die Zuweisung der Bezeichnung der Verknüpfung bzw. des Objekts zu der Reihe vorgenommen werden.
- 20 Das erfindungsgemäße Verfahren erzeugt also eine Zwischenformattabelle, ähnlich der marktüblicher Tabellenverarbeitungsprogramme, bei der einer ersten Dimension, beispielsweise in Querrichtung, Bezeichnungen für identifizierte Objekte zugewiesen werden. Solche Reihen können in diesem Fall als Spalten bezeichnet werden. Jede Spalte dient also zur Aufnahme
- 25 von Schnittstellendefinitionen eines Objekts. In der zweiten Dimension, beispielsweise den Zeilen einer Tabelle, werden die identifizierten Verknüpfungen zwischen den Objekten aufgetragen. Es ergibt sich hierdurch ein im konkreten Fall
- 30 zweidimensionales Raster mit Zellen. Unter "Bezeichnungen" sind hierbei eindeutige Namensgebungen zu verstehen, die der späteren Wiederauffindung eines Objekts, einer Schnittstelle oder anderer Elemente dienen. Die Bezeichnungen können in willkürlich vom Umsetzungsprogramm gewählten Namen bestehen
- 35 oder in Numerierungen von 1 bis n für jedes der aufgefundenen identifizierten Objekte (wenn n die Zahl der insgesamt iden-

tifizierten Objekte); oder in einer anderen geeigneten Weise erfolgen.

Die Tabelle wird in der Rechenanlage in einer elektronischen, magnetischen, optischen oder sonstigen geeigneten Form in einem geeigneten Format gespeichert. Sie kann auch zur Dokumentation auf einem Drucker ausgegeben werden.

- Im einfachsten Fall handelt es sich um eine zweidimensionale Tabelle, die als Zwischenformat gewählt wird. Hiermit kann man die oben angegebenen Informationen über die Schnittstellendefinitionen darstellen und speichern. Es ist jedoch auch vorstellbar, weitere Informationen über die Schnittstellen zu gewinnen, beispielsweise über den zu einem Objekt gehörigen Typ der durch das Objekt simulierten Komponente, wenn es sich um eine Hardwarebeschreibungssprache handelt. Solche weiteren Eigenschaften von einzelnen Schnittstellen lassen sich in einer dritten oder weiteren Dimension darstellen.
- Die einfachsten Fälle von umsetzbarem Programmcode sind solche, bei denen lediglich ein Objekt im Programmcode identifiziert werden kann, das über eine oder mehrere externe Schnittstellen verfügt, sowie Programmcode, der zwei Objekte aufweist, die miteinander über eine interne Verknüpfung verbunden sind. Um diese beiden einfachsten Möglichkeiten umsetzen zu können, ist es notwendig, Alternativen im Verfahren vorzusehen, die entweder externe oder interne oder beide Arten von Schnittstellen identifizieren können. Auch bei der Besetzung der Zellen der Tabelle unter Zuweisung der Reihen müssen beide einfachste Möglichkeiten alternativ und kumulativ berücksichtigt werden.

Das erfindungsgemäße Verfahren führt zu einer Zwischenformattabelle, die alle Schnittstellendefinitionen, welche in einem Programmcode enthalten sind, in einer klar gegliederten und in einem, weiteren Umsetzungsprogrammen in einheitlicher Weise zugänglichen, Datenformat enthalten. Durch die zweidi-



mentionale Darstellung ist auch die visuelle Inspektion des erzeugten Zwischenformats und damit die Eingriffmöglichkeit eines Operators oder Programmierers in einfacherer Weise gegeben, als dies ohne Zwischenformat oder bei herkömmlicher Darstellung mit Blockdiagrammen möglich sein würde.

Das Verfahren kann weiter verfeinert werden, wenn zumindest eine externe Verknüpfung identifiziert worden ist. Vorzugsweise wird eine erste spezielle Reihe der ersten Dimension zur Angabe des Modus der externen Schnittstelle bei der zumindest einen identifizierten externen Verknüpfung verwendet, wobei jeweils in diejenigen Zellen, die am Kreuzungspunkt der ersten speziellen Reihe der ersten Dimension und der Reihen der zweiten Dimension mit den Bezeichnungen der zumindest einen identifizierten externen Verknüpfung liegen, Angaben über den Modus der externen Schnittstelle bei der zumindest einen identifizierten externen Verknüpfung eingesetzt werden.

Das Verfahren kann bezüglich des Modus dadurch gekennzeichnet sein, daß die externen Schnittstelle eine Eingangsschnittstelle, eine Ausgangsschnittstelle, eine bidirektionale Schnittstelle oder eine Schnittstelle unbestimmter Flußrichtung sein kann.

Eine solche Angabe kann beispielsweise durch einen Zahlencode geschehen, wobei 0 eine Eingangsschnittstelle, 1 eine Ausgangsschnittstelle oder umgekehrt bedeuten kann, oder durch die Verwendung geeigneter Ausdrücke wie Ein, Aus, Ein/Aus, "In", "Out" oder "Inout".

Zusätzlich können die Datentypen der zumindest einen identifizierten Schnittstelle festgestellt werden, wobei zumindest einer zweiten speziellen Reihe der ersten Dimension eine Angabe der Datentypen der zumindest einen identifizierten Schnittstelle zugewiesen wird und jeweils in diejenigen Zellen, die am Kreuzungspunkt der zweiten speziellen Reihe und der Reihen der zweiten Dimension mit den Bezeichnungen der

zumindest einen identifizierten Verknüpfung liegen, Bezeichnungen für die zu der zumindest einen identifizierten Verknüpfung gehörigen Datentypen eingesetzt werden.

- 5 Von Datentypen wird in der Datenverarbeitung allgemein gesprochen, um zu kennzeichnen, daß Werte von Variablen vom Compiler oder Interpreter in einer bestimmten Art und Weise interpretiert werden, um unterschiedliche Arten von Daten berücksichtigen zu können. Typische Datentypen sind binäre Daten, die nur die Werte 0 und 1 haben können, Integerzahlen, also ganze Zahlen, Floating-Point-Zahlen, bei denen zumindest eine Dezimalstelle vorhanden ist, "Character", bei denen Buchstaben oder Buchstabenfolgen bearbeitet werden etc. Diese Definition von Datentypen soll auch in der vorliegenden Erfindung verwendet werden. Im allgemeinen muß davon ausgegangen werden, daß eine identifizierte Schnittstelle nur einen Datentyp hat. Bei obiger Formulierung wurde jedoch der Plural gewählt, da zumeist mehrere identifizierte Schnittstellen vorhanden sein dürften. Genau wie oben ist auch hier die Bezeichnung als eine Namensgebung zu verstehen, wobei eine oder mehrere Namen für die Datentypen eingesetzt werden können. Auch hier kann die Bezeichnung als Zahlencode angegeben sein, wobei jede Zahl einen bestimmten Datentyp angeben soll, oder im Klartext, beispielsweise durch Angabe von "bit", "boolean", "character", "integer", etc., die unmittelbar in die Zellen hineingeschrieben werden können.

- Ein weiterer, von der Erfindung berücksichtigter Ansatz besteht darin, die Angabe des Datentyps einer Verknüpfung auch (oder ausschließlich) als Bestandteil der Bezeichnung einer Schnittstelle (extern oder intern) zu formulieren. Auf diese Möglichkeit wird weiter unten eingegangen werden.

- Schnittstellendefinitionen, insbesondere in Hardware-Beschreibungssprachen, können nicht nur die eigentlichen Schnittstellen, also die Verknüpfungen zwischen Objekten eines Programmcodes und/oder externen Objekten umfassen, son-

5 dern beschreiben auch die Angabe von innerhalb des Programmcodes verwendeten Konstanten, da diese für das Verhalten von Komponenten maßgeblich sein können, ohne innerhalb der Komponenten definiert werden zu müssen. Ihr natürlicher Ort ist daher in der Schnittstellendefinition. Im Sinne der vorliegenden Erfindung soll also auch die Konvertierung von Konstanten als eine Umsetzung von Schnittstellendefinitionen verstanden werden.

10 Im Sinne der vorliegenden Erfindung soll unter einer externen Konstanten eine solche verstanden werden, welche innerhalb eines Programmcodes zwar deklariert, jedoch nicht notwendigerweise definiert, also mit einem Zahlenwert versehen, wird. Die Einprägung der externen Konstanten mit einem Zahlenwert  
15 erfolgt i.A. von außerhalb des betrachteten Programmcodes liegenden Programmteilen. Eine solche externe Konstante wird zum Beispiel in VHDL als eine "generic" bezeichnet. Eine solche generic wird innerhalb einer sogenannten "entity" deklariert, das heißt jenem Bereich von Programmcode bei VHDL, in  
20 dem Schnittstellen einer Komponente deklariert werden (entity oder component). Unter einer internen Konstanten wird hingegen eine solche verstanden, die innerhalb des Programmcodes sowohl deklariert als auch definiert werden muß. Von Konstanten wird in der vorliegenden Erfindung allgemein gesprochen,  
25 wenn interne und/oder externe Konstanten in Frage kommen

Daher beinhaltet das erfindungsgemäße Verfahren vorzugsweise, daß zusätzlich zumindest eine interne Konstante in dem zumindest einen identifizierten Objekt und/oder zumindest eine externe Konstante, die von allen Objekten des Quellprogrammcodes verwendbar ist, identifiziert wird, der Datentyp der zumindest einen identifizierten internen Konstanten und/oder externen Konstanten festgestellt wird, zumindest einer dritten speziellen Reihe der ersten Dimension eine Angabe  
30 des Datentyps der zumindest einen Konstanten zugewiesen wird, zumindest einer ersten speziellen Reihe der zweiten Dimension Bezeichnungen der zumindest einen identifizierten internen

- Konstanten und/oder externen Konstanten zugewiesen werden und jeweils in diejenigen Zellen, die am Kreuzungspunkt der zumindest einen dritten speziellen Reihe und der zumindest einen ersten speziellen Reihe der zweiten Dimension mit Bezeichnungen der zumindest einen identifizierten Konstanten liegen, Bezeichnungen für den zu der zumindest einen identifizierten Konstanten gehörigen Datentyp eingesetzt werden.
- Auch hier wird jeder der identifizierten Konstanten eine der ersten speziellen Reihen der zweiten Dimension zugewiesen, um eine eindeutige Zuordnung der in der Tabelle erfassten Größen zu den Konstanten (wie im übrigen auch den Verknüpfungen) herstellen zu können.
- Auf diese Weise ist es möglich, bei späterer Umsetzung der Konstanten in einem Zielprogrammcode eine korrekte Konstantendefinition und damit korrekte Interpretation eines zur Konstanten gehörigen Werts zu ermöglichen.
- Zur exakten Angabe von Konstanten gehört, sofern das anhand des vorliegenden Programmcodes möglich ist, sinnvollerweise auch deren Wert bzw., sofern vorhanden, eine Berechnungsweise der Konstanten.
- Unter einer Berechnungsweise einer Konstanten wird dabei ein Programmcode verstanden, durch dessen Abarbeitung während des Programmlaufs der einer Konstanten zuzuweisende Wert ermittelt wird. Die Berechnung einer Konstanten ist lediglich eine komplexere Weise, der Konstanten einen Wert zuzuweisen.
- Daher ist das Verfahren vorzugsweise auch darauf gerichtet, daß zusätzlich einen Wert oder eine Berechnungsweise der zumindest einen identifizierten Konstanten und/oder externen Konstanten festgestellt wird; zumindest einer vierten speziellen Reihe der ersten Dimension die Angabe des Werts oder der Berechnungsweise der zumindest einen Konstanten zugewiesen wird, und jeweils in diejenigen Zellen, die am Kreuzungs-

- 5 punkt der zumindest einen vierten speziellen Reihe und den  
ersten speziellen Reihen der zweiten Dimension mit Bezeich-  
nungen der zumindest einen identifizierten Konstanten liegen,  
der Wert oder die Berechnungsweise der zumindest einen iden-  
tifizierten Konstanten eingesetzt wird. Auch hier kann der  
Umfang der Zwischenformattabelle vorteilhafterweise dadurch  
begrenzt werden, daß die zumindest eine vierte spezielle Rei-  
he zugleich die zumindest eine erste spezielle Reihe (der er-  
sten Dimension) ist.
- 10 Alternativ können die Angaben über Wert oder Berechnungsweise  
einer Konstanten auch am Kreuzungspunkt der, der Konstanten  
zugewiesenen, ersten speziellen Reihe mit einer der Reihen  
der ersten Dimension mit Bezeichnungen für die identifizier-  
ten Objekte eingesetzt werden. Auf diese Weise kann eine bes-  
sere Zuordnung von Konstanten erzielt werden, die nicht in  
einem allgemeinen Teil des Quellprogrammcodes definiert wor-  
den sind, sondern die in einem bestimmten Programmcodeobjekt  
definiert werden. Auf diese Weise bleibt die Zuordnung einer  
20 Konstanten zu einem Programmcodeobjekt, in dem sie definiert  
worden ist, erhalten.
- 25 Neben den Konstanten können auch die identifizierten Verknüp-  
fungen einen Wert oder eine Berechnungsweise aufweisen, die  
diesen am Anfang eines Programmlaufs zugewiesen wird. Daher  
ist das erfindungsgemäße Verfahren vorzugsweise dadurch ge-  
kennzeichnet, daß zusätzlich ein Wert oder eine Berechnungs-  
weise der zumindest einen identifizierten Verknüpfung festge-  
stellt wird;
- 30 zumindest einer fünften speziellen Reihe der ersten Dimension  
die Angabe des Werts oder der Berechnungsweise der zumindest  
einen identifizierten Verknüpfung zugewiesen wird; und
- 35 wobei jeweils in diejenigen Zellen, die am Kreuzungspunkt der  
zumindest einen fünften speziellen Reihe und einer der Reihen  
der zweiten Dimension mit der Bezeichnung einer identifizier-

ten internen Verknüpfung und/oder der Reihen der zweiten Dimension mit der Bezeichnung einer identifizierten externen Verknüpfung liegen, der Wert oder die Berechnungsweise der zumindest einen identifizierten Verknüpfung eingesetzt wird.

5

Im erfindungsgemäßen Verfahren, wie es bislang definiert und skizziert worden ist, sind Objekten, Verknüpfungen, Konstanten und Schnittstellen Bezeichnungen zugewiesen worden und diese sind teilweise auch in Zellen eingesetzt worden, ohne  
10 daß angegeben worden ist, wie die Bezeichnungen generiert werden.

Es ist möglich, daß die Bezeichnungen vom eine Umsetzung vornehmenden Programm automatisch nach internen Kriterien zugewiesen werden.  
15

Bevorzugt ist allerdings ein Verfahren, das den weiteren Schritt aufweist:

20 Identifizieren der Originalbezeichnungen, welche das zumindest eine Objekt, die zumindest eine Verknüpfung und/oder die zumindest eine Konstante im Quellprogrammcode aufweisen, wobei speziellen Titelreihen Angaben über die identifizierten Originalbezeichnungen zugewiesen werden, und die Originalbezeichnungen in Zellen der Titelreihen eingesetzt werden.  
25

Dieses Einsetzen erfolgt sinnvollerweise natürlich so, daß es jeweils in die Zellen erfolgt, die am Kreuzungspunkt der Titelreihen mit den Objekt-, Verknüpfungs- oder Konstantenreihen, die zu dem jeweils bezeichnenden Objekt gehören, erfolgt.  
30

Die Zellen mit den Bezeichnungen der internen Schnittstellen bzw. der externen Schnittstellen können über eine einfache,  
35 namentliche Identifizierung der Schnittstelle hinaus auch weitere Informationen über die Schnittstelle enthalten.

Die Bezeichnungen der zumindest einen Schnittstelle können dementsprechend zusammengesetzt sein aus einem Identifizierer für die jeweilige Schnittstelle und aus zumindest einer Angabe, die ausgewählt ist aus einer Kennzeichnung des Modus der Schnittstelle, des Datentyps der Schnittstelle, eines Vorgabewerts und der Angabe einer auf die Schnittstelle anzuwendenden Datentypkonvertierungsfunktion. Auf diese Weise lassen sich zusätzlich zur namentlichen Identifizierung auch gleich weitere Informationen über die Schnittstelle der Zwischenformattabelle entnehmen.

Unter einem Identifizierer ist hier eine namentliche Kennzeichnung der Schnittstelle, beispielsweise ein willkürlich zugewiesener Name, eine Nummer oder eine andere, zur Unterscheidung von anderen Schnittstellen geeignete Namensgebung zu verstehen.

Der Modus einer Schnittstelle ist, wie oben definiert, eine Kennzeichnung des Verhaltens der Schnittstelle bezüglich Ein- und Ausgangs von Signalen bzw. Daten. Eine Schnittstelle kann hierbei einen Modus aufweisen, der sie zur Ausgangs-, zur Eingangsschnittstelle, zu einer bidirektionalen Schnittstelle oder zu einer Schnittstelle unbestimmter Flußrichtung macht.

Der Datentyp der Schnittstelle kann mit Bezug auf das oben Gesagte beispielsweise ein Binärwert, eine Integerzahl o.ä. sein.

Ein Vorgabewert ist ein Wert, den eine Schnittstelle annehmen soll, wenn kein anderer Wert definiert worden ist und das Programm sich im Ausgangszustand befindet.

Eine Datentypkonvertierungsfunktion wird benötigt, wenn zwei Schnittstellen miteinander verknüpft werden sollen, deren Datentypen voneinander abweichen. In einem solchen Fall verwandelt die Datentypkonvertierungsfunktion das Datum der Aus-

gangsschnittstelle in ein Datum, das dem vom Datentyp der Ringangsschnittstelle entspricht.

Die zusätzlich in der Bezeichnung der Schnittstellen angeba-  
5 ren Informationen können einzeln oder zu mehreren angegeben  
werden, wobei die einzelnen Angaben voneinander durch bei-  
spielsweise Kommata oder andere Trennzeichen voneinander ge-  
trennt werden können oder indem die Repräsentation der Anga-  
ben in einer anderen, für die automatische Verarbeitung ge-  
10 eigneter Form erfolgt.

Entsprechend wie für Objekte, Verknüpfungen oder Konstanten  
kann das erfindungsgemäße Verfahren auch den weiteren Schritt  
aufweisen:

15 Identifizieren der Originalbezeichnungen, welche die zumin-  
dest eine Schnittstelle im Quellprogrammcode aufweist;

20 Verwenden der Originalbezeichnungen als Identifizierer.

Das erfindungsgemäße Verfahren eignet sich grundsätzlich zur  
Umsetzung eines beliebigen Quellprogrammcodes in eine Zwi-  
schenformattabelle. Besonders bevorzugt ist jedoch der Quell-  
programmcode einer Hardware-Beschreibungssprache, insbesonde-  
25 re bei solchen Hardware-Beschreibungssprachen, bei denen die  
Schnittstellendefinitionen als ein eigenständiger Pro-  
grammcode ausgeführt sind. Bei herkömmlichen Programmierspra-  
chen oder bei Hardware-Beschreibungssprachen, bei denen eine  
solche Trennung nicht erfolgt, ist die Anwendung des erfin-  
30 dungsgemäßen Verfahrens grundsätzlich möglich. Es wächst je-  
doch der Komplexitätsgrad bei der Analyse des Quellpro-  
grammcodes. Demgemäß ist ein Objekt vorzugsweise eine  
Schnittstelleninstanz eines elektronischen Bauteils, das die-  
ses repräsentiert. Weiterhin repräsentiert die zumindest eine  
35 interne Verknüpfung ein Signal, das zwischen zwei elektroni-  
schen Bauteilen gegeben wird. Die zumindest eine externe Ver-



knüpfung kann vorzugsweise einen sogenannten Port repräsentieren.

Das bislang beschriebene Verfahren zur Umsetzung von Schnittstellendefinitionen führt zu einer Zwischenformattabelle, welche in der Regel lediglich Informationen über eine Ebene eines Quellprogrammcodes enthalten kann, das heißt über einander nebengeordnete Objekte, die beispielsweise als nebenläufige Prozesse in einer Hardwaresimulation virtuell gleichzeitig ausgeführt werden. Ein Quellprogrammcode kann jedoch auch einen höheren Komplexitätsgrad erreichen, indem mittels Verzweigungen oder anderer Referenzen der Code verschachtelt, beziehungsweise in verschiedene Ebenen gegliedert worden ist. Ein bereits genanntes Beispiel ist die Referenz auf Datentypkonvertierungsfunktionen als Teil der Bezeichnung von Schnittstellen, die bereits auf Programmcode außerhalb der eigentlichen Schnittstellendefinition verweisen kann. Des weiteren ist es möglich, daß Objekte im Quellprogrammcode wiederum durch Quellprogrammcode definiert worden sind, der separat von diesem gehalten wird.

Um das erfindungsgemäße Verfahren und die resultierenden Zwischenformattabellen auch für solche Verweise zugänglich zu machen, kann das erfindungsgemäße Verfahren vorzugsweise dadurch gekennzeichnet sein, daß zumindest eines der identifizierten Objekte in sich einen Subquellprogrammcode enthält, der ebenfalls in ein Zwischenformat umgesetzt werden kann, und daß nach erfolgter Umsetzung eines Subquellprogrammcodes in eine Subformattabelle ein Verweis auf die Subformattabelle in eine Zelle der zum umgesetzten Objekt gehörigen Reihe der ersten Dimension eingesetzt wird. Hierfür kann eine beliebige Zelle verwendet werden, der noch keine andere Funktion zugewiesen ist, oder es werden entsprechend in der anderen Dimension Reihen angehängt, so daß sich weitere Zellen ergeben, die für diesen Zweck genutzt werden können.

Auch kann das erfindungsgemäße Verfahren so modifiziert werden, daß ein Verweis auf zumindest ein als Quellprogrammcode als separate Einheit gespeichertes, identifiziertes Objekt in eine Zelle der zum gespeicherten Objekt gehörenden Reihe der ersten Dimension eingesetzt wird. Hier wird also der Bereich der reinen Schnittstellendefinitionen verlassen, indem ein unmittelbarer Bezug zum Quellprogrammcode hergestellt wird. Dieser Verweis kann nützlich sein, wenn bei der Weiterverwandlung in den Zielprogrammcode nicht nur die Schnittstellendefinitionen vom Umformatierungsprogramm umgesetzt werden, sondern auch der die Objekte realisierende Programmcode. In diesem Fall kann das Programm durch die oben angegebenen Verweise einen unmittelbaren Zusammenhang zwischen den Schnittstellendefinitionen eines Objekts und seiner Funktionalität herstellen.

Um die Dimension der Zwischenformattabelle kompakt zu halten, kann es bevorzugt sein, bestimmte Reihen miteinander zu kombinieren, so daß anstelle von zwei Reihen nur noch eine verwendet werden muß. Diese Bündelung ist dort möglich, wo jeweils eine Reihe einer Dimension nur an solchen Zellen Einträge aufweisen, bei denen bei der anderen zu bündelnden Reihe keine Zellen mit Werten erscheinen. So bietet es sich beispielsweise an, daß die zumindest eine dritte spezielle Reihe zugleich die zumindest eine zweite spezielle Reihe ist. Es wird in diesem Fall also eine Reihe der ersten Dimension verwendet, um darin unterschiedliche Größen unterzubringen. Dies kann geschehen, indem die Reihe unterteilt wird, so daß beispielsweise zwei verschiedene untereinander stehende Bereiche der Reihe die unterschiedlichen Funktionen erfüllen, beziehungsweise im Falle einer Nebeneinanderreihung der einzelnen Zellen nebeneinanderliegende Bereiche der Reihe. Weitere Beispiele für eine Bündelung sind die Kombination der vierten speziellen Reihe mit der ersten speziellen Reihe oder der vierten speziellen Reihe mit der fünften speziellen Reihe.

Nach erfindungsgemäßer Umsetzung der Schnittstellendefinitionen eines Quellprogrammcodes in eine Zwischenformattabelle liegt eine solche Tabelle in der Rechenanlage vor, die nach Ausdruck oder durch Anzeige an einem Datensichtgerät einfach zu lesen, visuell zu inspizieren, gegebenenfalls durch Eingriff eines Programmierers händisch korrigier- oder abänderbar ist und die erfindungsgemäß als Ausgangsbasis für eine Schnittstellendefinition eines Zielprogrammcodes dienen kann. Bezüglich der Vorteile, Effekte und Aspekte des Verfahrens zur Umsetzung einer Zwischenformattabelle in einen Zielprogrammcode wird auf das oben Gesagte verwiesen und vollinhaltlich Bezug genommen.

Demgemäß ist die Erfindung ebenfalls gerichtet auf ein Verfahren zur Umsetzung von Schnittstellendefinitionen aus einer solchen zumindest zweidimensionalen Zwischenformattabelle mit in einer ersten Dimension angeordneten ersten Reihen, in einer zweiten Dimension angeordneten zweiten Reihen und Zellen an den Kreuzungspunkten der ersten und zweiten Reihen, bei der Reihen der ersten Dimension Bezeichnungen für zumindest ein Objekt zugewiesen sind, bei der Reihen der zweiten Dimension Bezeichnungen für zumindest eine interne Verknüpfung zwischen den Objekten und/oder zumindest eine externe Verknüpfung eines Objekts zugewiesen sind, und bei der jeweils in diejenigen Zellen, die am Kreuzungspunkt einer der Reihen der ersten Dimension mit der Bezeichnung eines Objekt und einer der Reihen der zweiten Dimension mit der Bezeichnung einer internen Verknüpfung liegen, Bezeichnungen für eine zu sowohl dem jeweiligen Objekt als auch der Verknüpfung gehörigen internen Ausgangsschnittstelle und/oder internen Eingangsschnittstelle eingesetzt sind und/oder bei der jeweils in diejenigen Zellen, die am Kreuzungspunkt einer der Reihen der ersten Dimension mit der Bezeichnung eines Objekts und einer der Reihen der zweiten Dimension mit der Bezeichnung einer externen Verknüpfung liegen, Bezeichnungen für die zu sowohl dem jeweiligen Objekt als auch der externen Verknüpfung gehörigen externen Ausgangsschnittstelle und/oder exter-

nen Eingangsschnittstelle eingesetzt sind, mittels einer das Verfahren ausführenden Rechenanlage in einen Zielprogrammcode mit den nachfolgend beschriebenen Schritten.

- 5 Durch die obige Definition wird eine elementare Zwischenformattabelle spezifiziert. Dieses erfindungsgemäße Verfahren, weist folgende Schritte auf:

- 10 Erstellen von zumindest einem Programmcode-Objekt anhand der in der Zwischenformattabelle enthaltenen Informationen über das zumindest eine Objekt;

- 15 Zuweisen von zugehörigen internen Ausgangsschnittstellen und/oder internen Eingangsschnittstellen zu ihren Programmcode-Objekten;

- 20 Erstellen zumindest einer Verknüpfung zwischen Programmcode-Objekten anhand der in der Zwischenformattabelle enthaltenen Informationen über die internen Verknüpfungen der internen Eingangsschnittstellen und internen Ausgangsschnittstellen, und/oder

- 25 Zuweisen von zugehörigen externen Ausgangsschnittstellen und/oder externen Eingangsschnittstellen zu ihren Programmcode-Objekten.

- 30 Unter einem Programmcode-Objekt im Sinne der vorliegenden Erfindung ist ein Objekt, geschrieben in einem Programmcode, zu verstehen, das als Teil des Zielprogrammcodes durch das erfindungsgemäße Verfahren generiert wird.

- 35 Wie schon beim erfindungsgemäßen Verfahren zur Umsetzung von Schnittstellendefinitionen im Rahmen von Quellprogrammcodes in ein Zwischenformat ist auch bei diesem erfindungsgemäßen Verfahren dem Fachmann geläufig, wie er die in der Zwischenformattabelle enthaltenen Informationen so exzerpieren kann, daß ein Umsetzen in einen Zielprogrammcode möglich ist.

Beispielsweise kann das dafür geeignete automatische Programm zunächst in den Reihen der ersten Dimension anhand der vorgegebenen Bezeichnungen die einzelnen definierten Objekte ermitteln und dann innerhalb der Reihen Konstanten sowie interne und externe Verknüpfungen ermitteln und diesen die an den Kreuzungspunkten genannten Schnittstellen zuweisen.

Vorzugsweise ist zumindest einer zweiten speziellen Reihe der ersten Dimension der Zwischenformattabelle eine Angabe der Datentypen der zumindest einen Schnittstelle zugewiesen und sind jeweils in diejenigen Zellen, die am Kreuzungspunkt der zweiten speziellen Reihe und Reihen der zweiten Dimension mit den Bezeichnungen der zumindest einen Verknüpfung liegen, Bezeichnungen für die zur zumindest einen Verknüpfung gehörigen Datentypen eingesetzt sind, wobei diese Weiterentwicklung den weiteren Schritt aufweist: Definieren der Datentypen der dem zumindest einen Programmcode-Objekt zugewiesenen zu der zumindest einen Verknüpfung gehörigen zumindest einen Schnittstelle.

Wie oben erläutert, ist unter einer Schnittstellendefinition im Sinne der vorliegenden Erfindung auch die Definition von Konstanten, die von verschiedenen Objekten verwendet werden, zu verstehen. Daher kann das erfindungsgemäße Verfahren vorzugsweise dadurch gekennzeichnet sein, daß zumindest einer dritten speziellen Reihe der ersten Dimension der Zwischenformattabelle eine Angabe der Datentypen von zumindest einer Konstanten in zumindest einem Objekt und/oder zumindest einer externen Konstanten, die von allen Objekten verwendbar ist, zugewiesen ist, zumindest einer ersten speziellen Reihe der zweiten Dimension Bezeichnungen der zumindest einen Konstanten und/oder zumindest einen externen Konstanten zugewiesen sind und jeweils in diejenigen Zellen, die am Kreuzungspunkt der zumindest einen dritten speziellen Reihe und der zumindest einen ersten speziellen Reihe der zweiten Dimension mit Bezeichnungen der zumindest einen Konstanten liegen, Bezeich-

nungen für die zu der zumindest einen Konstanten gehörigen Datentypen eingesetzt werden; mit dem weiteren Schritt: Definieren von zumindest einer internen Konstante und/oder von zumindest einer externen Konstante in dem zumindest einen  
5 Programmcode-Objekt und/oder in einem allgemeinen Teil des Zielprogrammcodes.

Weiterhin können zumindest einer vierten speziellen Reihe der ersten Dimension der Zwischenformattabelle Angaben über einen  
10 Wert oder eine Berechnungsweise der zumindest einen internen Konstanten und/oder der zumindest einen externen Konstanten zugewiesen sein und jeweils in diejenigen Zellen, die am Kreuzungspunkt der vierten speziellen Reihe mit den ersten speziellen Reihen der zweiten Dimension mit Bezeichnungen der  
15 zumindest einen Konstanten liegen, der Wert oder die Berechnungsweise der zumindest einen Konstanten eingesetzt sein, wobei diese Weiterentwicklung den weiteren Schritt aufweist: Zuweisen des Werts oder der Berechnungsweise der zumindest einen Konstanten zu der zumindest einen im Programmcode definierten Konstanten.  
20

Das erfindungsgemäße Verfahren kann weiterhin dadurch gekennzeichnet sein, daß zumindest einer fünften speziellen Reihe der ersten Dimension der Zwischenformattabelle eine Angabe  
25 über den Wert oder die Berechnungsweise der zumindest einen Verknüpfung zugewiesen ist; und jeweils in diejenigen Zellen, die am Kreuzungspunkt der zumindest einen fünften speziellen Reihe und einer der Reihen der zweiten Dimension mit der Bezeichnung einer internen Verknüpfung und/oder der Reihen der zweiten Dimension mit der Bezeichnung einer externen Verknüpfung liegen, der Wert oder die Berechnungsweise der zumindest einen Verknüpfung eingesetzt ist;  
30

mit dem weiteren Schritt:  
35

-Zuweisen des Werts oder der Berechnungsweise der zumindest einen Verknüpfung zu der im Zielprogrammcode erstellten Verknüpfung.

- 5 Weiterhin können vorzugsweise speziellen Titelreihen der Zwischenformattabelle Angaben über die Bezeichnungen des zumindest einen Objekts der zumindest einen Verknüpfung und/oder der zumindest einen Konstanten zugewiesen sein, wobei die Bezeichnungen in Zellen der Titelreihen eingesetzt sind und das
- 10 Verfahren den weiteren Schritt aufweist: Benennen des zumindest einen Programmcode-Objekts, der zumindest einen Verknüpfung und/oder der zumindest einen Konstanten anhand der Bezeichnungen in den Zellen der Titelreihen der Zwischenformattabelle.
- 15 Somit erhalten die entsprechenden Objekte, Verknüpfungen und/oder Konstanten im Zielprogrammcode die gleichen Bezeichnungen, die sie bereits im Quellprogrammcode vor Umsetzung in eine Zwischenformattabelle hatten.
- 20 Das erfindungsgemäße Verfahren kann weiterhin dadurch gekennzeichnet sein, daß zumindest in eine Zelle einer zu einem Objekt gehörenden Reihe der ersten Dimension ein Verweis auf eine Subformattabelle eingesetzt ist, mit dem weiteren
- 25 Schritt: Verbinden des aus dem Objekt erzeugten Programmcode-Objekts mit aus der Subformattabelle erzeugten Subprogrammcode.
- 30 Das Verfahren kann auch dadurch gekennzeichnet sein, daß zumindest in eine Zelle der zu einem Objekt gehörenden Reihe der ersten Dimension ein Verweis auf als separate Einheit gespeicherten Quellprogrammcode eingesetzt ist, mit dem weiteren Schritt: Verbinden des aus dem zumindest einen Objekt erzeugten Programmcodeobjekts mit dem als separate Einheit gespeicherten Quellprogrammcode.
- 35

Wie bereits in Bezug auf das Verfahren zur Umsetzung von Programmcode in eine Zwischenformattabelle ausgeführt, kann es sinnvoll sein, um die Dimension der Zwischenformattabelle kompakt zu halten, bestimmte Reihen miteinander zu kombinieren, so daß anstelle von zwei Reihen nur noch eine verwendet werden muß. Auch diese Bündelung kann bei der Umsetzung in einen Zielprogrammcode berücksichtigt werden.

10 Zur vorliegenden Erfindung gehören nicht nur die Verfahren zur Umsetzung von Programmcode, sondern auch die spezifische Ausgestaltung der Zwischenformattabelle. Beispielsweise enthalten die Reihen der Tabelle alle Signale und Konstanten der Architektur und alle Ports und Generics der Entity, sofern es sich um Repräsentationen in einer Hardware-

15 Beschreibungssprache handelt. In den Reihen der verbleibenden Dimension sind die nebenläufigen Statements, wie beispielsweise Komponenten-Instantiation, Prozeß, konkurrierende Signalzuweisung etc. aufgelistet. Die Zwischenformattabelle zeigt übersichtlich, welche Komponenten mit welchen Signalen

20 von welchem Typ verbunden sind und wie die Signalflußrichtung ist. Andererseits kann mit einem Blick festgestellt werden, welche Komponente mit einem bestimmten Signal verbunden ist, das heißt wer das Signal generiert und wer es einliest. Bezüglich der Vorteile, Effekte und Aspekte der Zwischenformattabelle der vorliegenden Erfindung wird auf das oben ge-

25 sagte verwiesen und vollinhaltlich Bezug genommen.

Demzufolge ist die Erfindung auch gerichtet auf eine Zwischenformattabelle zur Speicherung von in einem Programmcode enthaltenen Schnittstelleninformationen in einer Rechenanlage

30 mit zumindest zwei Dimensionen; mit in einer ersten Dimension angeordneten Reihen, in einer zweiten Dimension angeordneten Reihen und Zellen an den Kreuzungspunkten; wobei Reihen der ersten Dimension Bezeichnungen für zumindest ein Objekt in

35 dem Programmcode zugewiesen sind, wobei Reihen der zweiten Dimension Bezeichnungen für zumindest eine interne Verknüpfung zwischen Objekten und/oder Bezeichnungen für zumindest



eine externe Verknüpfung des Programmcodes zugewiesen sind und wobei jeweils in diejenigen Zellen, die am Kreuzungspunkt einer der Reihen der ersten Dimension mit der Bezeichnung eines Objekts und einer der Reihen der zweiten Dimension mit der Bezeichnung einer internen Verknüpfung liegen, Bezeichnungen für eine zu sowohl dem jeweiligen Objekt als auch der internen Verknüpfung gehörigen Ausgangsschnittstelle und/oder Eingangsschnittstelle eingesetzt sind und/oder jeweils in diejenigen Zellen, die am Kreuzungspunkt einer der Reihen der ersten Dimension mit der Bezeichnung eines Objekts und einer der Reihen der zweiten Dimension mit der Bezeichnung einer externen Verknüpfung liegen, Bezeichnungen für die zu sowohl dem jeweiligen Objekt als auch der externen Verknüpfung gehörigen Ausgangsschnittstelle und/oder Eingangsschnittstelle eingesetzt sind.

Vorzugsweise ist eine erste spezielle Reihe der ersten Dimension zur Angabe des Modus einer externen Schnittstelle bei der zumindest einen externen Verknüpfung verwendet. Dabei sind jeweils in diejenigen Zellen, die am Kreuzungspunkt der ersten speziellen Reihe der ersten Dimension und der Reihen der zweiten Dimension mit den Bezeichnungen der zumindest einen externen Verknüpfung liegen, Angaben über den Modus der externen Schnittstelle bei der zumindest einen externen Verknüpfung eingesetzt. Wie bereits oben gilt auch hier, daß eine Schnittstelle eine Eingangs-, eine Ausgangsschnittstelle, eine bidirektionale Schnittstelle oder eine Schnittstelle unbestimmter Flußrichtung sein kann.

Weiterhin kann in der Zwischenformattabelle zumindest einer zweiten speziellen Reihe der ersten Dimension eine Angabe der Datentypen der zumindest einen Schnittstelle zugewiesen sein und jeweils in diejenigen Zellen, die am Kreuzungspunkt der zweiten speziellen Reihe und der Reihen der zweiten Dimension mit den Bezeichnungen der zumindest einen Verknüpfung liegen, Bezeichnungen für die zu der zumindest einen Verknüpfung gehörigen Datentypen eingesetzt sein. Unter einer Verknüpfung

sind hier sowohl interne als auch externe Verknüpfungen subsumiert.

- Weiterhin enthält die Zwischenformattabelle vorzugsweise zu-
- 5    mindest eine dritte spezielle Reihe der ersten Dimension, der eine Angabe der Datentypen zumindest einer internen Konstanten und/oder zumindest einer externen Konstanten aus dem Programmcode zugewiesen ist, erste spezielle Reihen der zweiten Dimension, der Bezeichnungen der zumindest einen internen
- 10    Konstanten und/oder der zumindest einen externen Konstanten zugewiesen sind, wobei jeweils in diejenigen Zellen, die am Kreuzungspunkt der dritten speziellen Reihe und der ersten speziellen Reihen der zweiten Dimension mit Bezeichnungen der zumindest einen Konstanten liegen, Bezeichnungen für die zu
- 15    der zumindest einen Konstanten gehörigen Datentypen eingesetzt sind. Auch hier gilt vorzugsweise, daß die zumindest eine dritte spezielle Reihe zugleich die zumindest eine spezielle zweite Reihe ist.
- 20    In der Zwischenformattabelle kann weiterhin zumindest einer vierten speziellen Reihe der ersten Dimension die Angabe eines Werts oder einer Berechnungsweise der zumindest einen Konstanten zugewiesen sein und können jeweils in die diejenigen Zellen, die am Kreuzungspunkt der zumindest einen vierten
- 25    speziellen Reihe und der ersten speziellen Reihe der zweiten Dimension mit Bezeichnungen der zumindest einen Konstanten liegen, der Wert oder die Berechnungsweise der zumindest einen Konstanten eingesetzt sein.
- 30    Die erfindungsgemäße Zwischenformattabelle kann dadurch gekennzeichnet sein, daß zumindest einer fünften speziellen Reihe der ersten Dimension die Angabe eines Werts oder einer Berechnungsweise der zumindest einen Verknüpfung zugewiesen ist; und
- 35    jeweils in diejenigen Zellen, die am Kreuzungspunkt der zumindest einen fünften speziellen Reihe und einer der Reihen

der zweiten Dimension mit der Bezeichnung einer internen Verknüpfung und/oder der Reihen der zweiten Dimension mit der Bezeichnung einer externen Verknüpfung liegen, der Wert oder die Berechnungsweise der zumindest einen Verknüpfung eingesetzt ist.

Auch für die erfindungsgemäße Zwischenformattabelle gilt, daß speziellen Titelreihen Angaben über die Originalbezeichnungen zugewiesen sein können, welche das zumindest eine Objekt, die zumindest eine Verknüpfung und/oder die zumindest eine Konstante im Programmcode aufweisen, und die Originalbezeichnungen in Zellen der Titelreihen eingesetzt sind.

Unter Programmcode ist hier ein Quellprogrammcode zu verstehen, der mit der Zwischenformattabelle dadurch korreliert ist, daß sie aus ihm generiert worden ist. Auch können die Bezeichnungen der zumindest einen Eingangsschnittstelle und/oder der zumindest einen Ausgangsschnittstelle zusammengesetzt sein aus einem Identifizierer für die jeweilige Schnittstelle und aus zumindest einer Angabe, die ausgewählt ist aus einer Kennzeichnung des Modus der Schnittstelle, des Datentyps der Schnittstelle, eines Vorgabewerts und der Angabe einer auf die Schnittstelle anzuwendenden Datentypkonvertierungsfunktion.

Die erfindungsgemäße Zwischenformattabelle kann weitere Informationen enthalten. So ist es üblich, bei Hardwarebeschreibungssprachen Kommentare zu verwenden, um Sachverhalte zu erläutern, aber auch, um Analysewerkzeugen oder Syntheseprogrammen Anweisungen zu übermitteln die außerhalb des eigentlichen Programmcodes liegen (sog. Pragmas) und für die in der verwendeten Programmiersprache keine Konstrukte vorgesehen sind. Auch Informationen über das Zeitverhalten eines Programmes, das ebenfalls nur in Kommentarform im Programmcode vorkommen kann, kann auf diese Weise mit umgesetzt oder von entsprechenden Werkzeugen verwendet werden. Das die Zwischenformattabelle erzeugende Umsetzungsprogramm kann zu-

dem weitere Informationen hinzufügen, die sich durch die Analyse des Ausgangsprogrammcodes ergeben haben und die helfen können, Ambiguitäten in der Zwischenformattabelle aufzulösen. Solche und andere Informationen können auch vom Umsetzungsprogramm, welches die Zwischenformattabelle in Programmcode rückverwandelt, als Anweisungen zur Steuerung der Umsetzungsabläufe verwendet werden. Auch die Verwendung solcher Informationen durch einen Anwender ist denkbar. Im folgenden werden alle oben dargestellten Informationen als Annotationen bezeichnet.

Die erfindungsgemäße Zwischenformattabelle kann daher dadurch gekennzeichnet sein, daß beliebige Zellen der Zwischenformattabelle Annotationen enthalten können, welche der Steuerung von Programmen zu Analyse der in der Zwischenformattabelle enthaltenen Informationen und/oder zur Information eines Anwenders dienen können.

Die praktische Umsetzung von Annotationen kann dadurch erreicht werden, daß zumindest eine weitere Dimension der Zwischenformattabelle den Annotationen zugeordnet ist, wobei Reihen der weiteren Dimension bestimmte Typen von Annotationen zugewiesen sind und eine einzusetzende Annotation an den Kreuzungspunkten der Reihen der ersten und der zweiten Dimensionen, die für die Annotation maßgeblich sind, mit der Reihe der weiteren Dimension, die dem einzusetzenden Typ von Annotation zugewiesen ist, die Annotation eingesetzt wird.

Die erfindungsgemäße Zwischenformattabelle kann entweder bei ihrer Erzeugung oder im späteren Verlauf der Bearbeitung in vielfältiger Weise modifiziert werden. Zur Steigerung der Übersichtlichkeit können Signale beziehungsweise Verknüpfungen zu Gruppen zusammengefaßt werden, die immer in der gleichen Weise an den Schnittstellen der Objekte (z. B. nebenläufigen Blöcke) verwendet werden. Für diese Signalgruppen, sogenannte "bundles" gibt es beispielsweise in einer Hardware-Programmiersprache, wie VHDL, keine Entsprechung, so daß von

einer VHDL-Beschreibung entsprechend abstrahiert werden könnte. Auch sogenannte Concurrent Statements (die alternativ zum Einsatz kommen) könnten in eine gemeinsame Spalte gruppiert werden. Bei Verwendung von VHDL entspricht dies einem sogenannten Block.

Durch interaktives Umsortieren von Reihen der ersten Dimension und Reihen der zweiten Dimension kann die Nutzbarkeit der Tabelle weiter erhöht werden, beispielsweise können nach der Selektion einer Reihe der zweiten Dimension die zugehörigen Spalten direkt benachbart angeordnet werden, um die Verbindungen des selektierten Objekts besser analysieren zu können. Auch weitere Sortierkriterien sind auf eine erfindungsgemäße Zwischenformattabelle anwendbar, beispielsweise eine alphabetische Anordnung, eine Anordnung, die nach einem Präfix oder einem Postfix sortiert ist, eine Anordnung nach der Betriebsart der externen Verknüpfungen, eine Anordnung nach dem verwendeten Datentyp der Verknüpfung oder einer Reihenfolge wie sie im Originalquellprogrammcode vorgekommen ist. Daher ist es möglich, bei der erfindungsgemäßen Tabelle Reihen interaktiv ausblenden zu lassen, um die Übersichtlichkeit zu erhöhen.

In der Zwischenformattabelle können auch Manipulationen vorgenommen werden, die zu einer Codetransformation bei der späteren Umsetzung in einen Zielprogrammcode führen würden. Beispiele für solche Transformationen können sein:

Gruppieren, das heißt Erzeugen einer neuen Tabelle mit gruppierten Objekten und Zusammenfassung der damit assoziierten Originalreihen zu einer neuen Reihe,

Umgruppieren, da heißt Ersetzen einer Originalreihe durch die Reihen einer weiteren Zwischenformattabelle einer niedrigeren Hierachiestufe, auf die ursprünglich die Originalreihe verwiesen hat.

Im folgenden soll an zwei Beispielen dargelegt werden, wie die Umsetzung von Programmcode in eine Zwischenformattabelle erfolgen kann. Hierbei wird Bezug auf die Zeichnungen genommen, in denen folgendes dargestellt ist:

5

Figur 1 zeigt eine Ausführungsform der Zwischenformattabelle der vorliegenden Erfindung, die aus einem ersten Beispielprogramm erzeugt worden ist; und

- 10    Figur 2 zeigt eine weitere Ausführungsform der Zwischenformattabelle der vorliegenden Erfindung, die aus einem zweiten Beispielprogramm erzeugt worden ist.

Das folgende erste, einfache Beispielprogramm:

15

```
entity top is
```

```
  port ( a : in bit;
```

```
         b : in integer;
```

```
         c : in boolean;
```

20

```
         d : out integer );
```

```
end top;
```

```
architecture arch of top is
```

```
  component comp_1
```

25

```
    port ( e : in bit;
```

```
          f : in integer;
```

```
          g : in bit;
```

```
          d : out integer );
```

```
  end component;
```

30

```
  component comp_2
```

```
    port ( i : in integer;
```

```
          j : in boolean;
```

```
          k : out bit;
```

```
          l : out integer );
```

35

```
  end component;
```

```
  signal si : integer;
```

```
  signal sb : bit;
```

```
begin
  inst_1 : comp_1
    port map( e => a,
              f => b,
5           g => sb,
              h => si );
  inst_2 : comp_2
    port map( i => si,
              j => c,
10           k => sb,
              l => d );
end arch;
```

15 zeigt zwei Programmobjekte inst\_1 und inst\_2 vom Typ comp\_1 und comp\_2 mit je vier Schnittstellen. Die beiden Objekte sind miteinander durch die zwei Verknüpfungen (Signale) si und sb über die Schnittstellen h und i beziehungsweise g und k miteinander verbunden.

20 Darüber hinaus weist inst\_1 noch zwei externe Schnittstellen f und e auf, die von den externen Verknüpfungen b und a bedient werden; inst\_2 die externe Schnittstelle j der Verknüpfung c und l der Verknüpfung d.

25 Im Beispielprogrammcode werden diese Schnittstellen definiert, zusammen mit ihrer Funktion als Eingangs- oder Ausgangsschnittstelle und den bei ihnen verwendeten Datentypen.

30 Figur 1 zeigt eine nach Anwendung des erfindungsgemäßen Verfahrens zur Umsetzung von Schnittstellen-Definitionen in ein Zwischenformat sich ergebende Zwischenformattabelle 1. Die beispielhafte Zwischenformattabelle 1 weist eine erste Dimension in Richtung des Pfeils 2 und eine zweite Dimension in Richtung des Pfeils 3 auf. Zunächst werden zwei Reihen erster  
35 Dimension 4 definiert, denen die Bezeichnungen der Objekte inst\_1 und inst\_2 zugewiesen werden. In der zweiten Dimension 3 werden Reihen zweiter Dimension 5 für interne Verknüpfungen

zugewiesen. Nunmehr werden in die sich ergebenden Kreuzungszellen 7 Bezeichnungen 8 für die verschiedenen internen Schnittstellen eingetragen. Zur Verbesserung der Übersichtlichkeit können den Reihen der ersten und zweiten Dimension außerdem Bezeichnungen explizit beigeordnet werden. Dies geschieht mit Hilfe von Titelreihen 24 beziehungsweise 25. In Titelreihe 24 werden in dafür vorgesehene Zellen für die Bezeichnung eines Objekts 29 Bezeichnungen für die Objekte 26 eingetragen. In der Titelreihe 25 werden in dafür vorgesehenen Zellen für Bezeichnungen einer internen Verknüpfung 31 diese Bezeichnungen für interne Verknüpfungen 28 eingetragen. Wie ersichtlich ist, sind in die Zellen 7 Bezeichnungen 8 eingetragen, die einen Identifizierer 32 einer internen Schnittstelle enthalten, nämlich deren Originalbezeichnung im Beispielprogrammcode sowie eine Kennzeichnung, ob es sich um eine Ausgangsschnittstelle (out) oder eine Eingangsschnittstelle (in) handelt. Weiter sind in der Tabelle Reihen zweiter Dimension 6 für externe Verknüpfungen vorgesehen. An den Kreuzungspunkten dieser Reihen 6 mit den Reihen der ersten Dimension 4 werden die für externe Verknüpfungen verwendeten Schnittstellen mit ihren Bezeichnungen 10 in Zellen 9 eingetragen. Auch hier besteht die Bezeichnung aus einem Identifizierer 33 sowie einer Angabe über den Character der Schnittstellen als Ein- oder Ausgabeschnittstellen. In die Titelreihen 25 werden die Originalbezeichnungen 27 der externen Verknüpfungen in Zellen 30 eingetragen. Zusätzlich sind in einer ersten speziellen Reihe 11 an den Kreuzungspunkten mit den Reihen der zweiten Dimension für externe Verknüpfungen 12 Angaben für das Vorliegen einer Eingangsschnittstelle oder einer Ausgangsschnittstelle bei den externen Verknüpfungen eingetragen.

In der Beispielszwischenformattabelle 1 sind auch die Datentypen der Verknüpfungen angegeben. Hierzu wird eine zweite spezielle Reihe 14 verwendet, in deren Zellen 15 an den Kreuzungspunkten zu den Reihen zweiter Dimension für externe Verknüpfungen 6, aber auch zu den Reihen zweiter Dimension für



interne Verknüpfungen 5 jeweils eine Angabe 16 darüber gemacht ist, von welchem Datentyp die jeweilige Verknüpfung ist. Hierfür werden die in der Datenverarbeitung üblichen Begriffe, wie bit, integer, boolean, etc. verwendet.

5

Der obere Bereich dieser zweidimensionalen Zwischenformattabelle 1 ist der Angabe von Konstanten des Beispielsprogrammcodes vorbehalten. Da im angegebenen Beispielprogrammcode keine Konstanten definiert sind, werden hier lediglich zur Verdeutlichung die Buchstaben x und y eingesetzt.

10 Zur Angabe der Konstanten werden erste spezielle Reihen der zweiten Dimension 18 sowie für die Angabe des Datentype eine dritte spezielle Reihe 17 in der ersten Dimension und für die Angabe des Werts der Konstanten eine vierte spezielle Reihe 15 21 der ersten Dimension verwendet. In die Kreuzungspunktzellen 19 der dritten speziellen Reihen 17 und der speziellen Reihen der zweiten Dimension 18 werden die Werte für externe (oben) und interne (unten) Konstanten x eingetragen, wie mit dem Bezugszeichen 20 kenntlich gemacht. Die Werte der vorhandene Konstanten werden in Zellen 22 als y (23) an den Kreuzungspunkten zwischen der vierten speziellen Reihe und den ersten speziellen Reihen der zweiten Dimension 18 eingetragen.

25 Das folgende zweite Beispielprogramm ist eine erweiterte Fassung des ersten Beispielprogramms.

entity top is

generic( y : integer := 3);  
30 port ( a : in bit;  
b : in integer;  
c : in boolean := true;  
d : out integer );

end top;

35

architecture arch of top is  
component comp\_1

```

        generic( x : integer := 4);
        port ( e : in bit;
              f : in integer := 7;
              g : in bit;
5          d : out integer );
    end component;
    component comp_2
        generic( z : integer);
        port ( i : in integer;
10         j : in boolean;
           k : out bit;
           l : out integer );
    end component;
    constant ci : integer := y + 5;
15    signal si_1, si_2 : integer;
    signal sb : bit := '1';
    function boolean2bit(arg : boolean) return bit is
    begin
        return bit'val(boolean'pos(arg));
20    end boolean2bit;
    begin
        p0 : si_2 <= si_1 + ci;
        inst_1 : comp_1
            port map( e => a,
25                f => b,
                g => sb,
                h => si_1 );
        inst_2 : comp_2
            generic map( z => y)
30        port map( i => si_2,
                j => boolean2bit(c),
                k => sb,
                l => d );
    end arch;
35

```

Hier werden verschiedene Konstanten im allgemeinen Teil des Programms und bei den einzelnen Programmobjekten definiert.

Außerdem wird neben den beiden Objekten inst\_1 und inst\_2 als Komponentenrepräsentanten auch ein weiteres Objekt p0 angelegt. Schließlich ist ebenfalls eine Datenkonvertierungsfunktion boolean2bit vorgesehen, welche die Eingangsdaten auf der externen Schnittstelle "j" umwandelt.

Figur 2 zeigt eine Zwischenformattabelle gemäß einer weiteren Ausführungsform der vorliegenden Erfindung, bei der ein gegenüber der Tabelle der Figur 1 abweichendes Darstellungsformat verwendet wurde. Die für Figur 1 verwendeten Bezugszeichen werden auch bezüglich Figur 2 mit gleichen Bedeutungen verwendet.

Insbesondere die Bündelung von speziellen Reihen der ersten Dimension ist hier anders durchgeführt worden. Die Tabelle ist weniger kompakt als die in Figur 1 dargestellte.

In Figur 2 sind nicht nur Werte 23 der Konstanten und Berechnungsweise 35 der Konstante ci in einer vierten speziellen Reihe 21 eingesetzt worden, sondern auch die Werte 36 von Schnittstellen in einer fünften speziellen Reihe 34. Die vierte spezielle Reihe 21 ist hierbei zugleich die fünfte spezielle Reihe 34.

Die in diesem Beispiel vorhandenen Konstanten werden mit ihren Bezeichnungen 38 in dafür vorgesehene Zellen 37 der speziellen Reihen 18 der zweiten Dimension 3, welche an der Kreuzung zur Titelreihe 25 liegen, eingesetzt. Eine externe Konstante x, welche erst innerhalb einer Komponentendeklaration deklariert (und sofort mit dem Wert "4" definiert) wird, wird mit ihrer Bezeichnung 40 in die Zelle am Kreuzungspunkt der Reihe 4 der ersten Dimension, welche dem Objekt zugewiesen ist, innerhalb dessen die Konstante x deklariert worden ist, mit einer der speziellen Reihen der zweiten Dimension, eingesetzt. Da in diesem Fall keine namentlich bezeichnete externe Konstante für den gesamten Programmcode angegeben ist, verbleibt der Eintrag in der zuständigen Zelle 37 leer.

Eine weitere externe Konstante z wird ebenfalls mit ihrer Bezeichnung 40 in eine Zelle 39 eingesetzt. Im Unterschied zur Konstanten x wird hier jedoch im Programmcode ein "Mapping" vorgenommen, daß der Konstanten z die Konstante y zuweist.

- 5 Daher wird die Konstante z in die gleiche spezielle Reihe 18 der zweiten Dimension eingesetzt, die auch der Konstanten y zugewiesen ist.

- 10 Weiterhin ist in diesem Beispiel ein Objekt p0 definiert worden, welches eine Manipulation an der Verknüpfung si\_2 vornimmt, die sich aus si\_1 ableitet (siehe obigen 2. Programmcode). Auch für dieses Objekt ist eine Reihe 4 vorgesehen, in der die Verwendung der Konstanten ci als Bezeichnung 40 sowie die Verbindung des Objekts p0 mit den Signalen si\_1  
15 als eine Schnittstellenangabe mit der Bezeichnung 41 und si\_2 als eine Berechnungsvorschrift mit der Bezeichnung 43.

- 20 Eine der Bezeichnungen der Schnittstellen, gekennzeichnet mit dem Bezugszeichen 42, zeigt eine komplexere Struktur als die anderen Bezeichnungen 8, 41, da in dieser zusätzlich zum Identifizierer der Schnittstelle und der Datenflußrichtung auch eine Angabe zum Datentyp (hier "bit") und zur anzuwendenden Datentypkonvertierungsfunktion gemacht ist.

- 25 Insgesamt ergibt sich in beiden Beispielen eine sehr übersichtlich gestaltete Zwischenformattabelle 1, die von einem automatischen Umsetzungsprogramm in leichter Art und Weise in Schnittstellendefinitionen eines Zielprogrammcodes umgewandelt werden und auch der manuellen Nachbearbeitung durch einen Programmierer/Modellierer oder Hardware-, beziehungsweise  
30 Software-Ingenieur zugänglich gemacht werden kann.

- 35 Beispielsweise kann durch Entfernen und Hinzufügen von Tabelleneinträgen die gewünschte Verbindungsstruktur hergestellt werden. Die Schnittstellen der aktuellen Hierarchiestufe können durch Deklaration von Reihen zu Port-Signalen spezifiziert werden. Eine Erstellung oder Modifikation konkurrieren-

der Objekte kann nach Selektion der entsprechenden Reihe in einem separaten Texteditor erfolgen.

Die erfindungsgemäße Zwischenformattabelle kann auch als solche verwendet werden, um eine Schaltungsplanung durchzuführen, die später zu einem Programmcode führen kann (sogenannter Top-Down-Entwurf). Hierbei können zunächst leere Komponenten durch Einfügen von Reihen, beispielsweise von Spalten generiert werden. Anschließend werden die Verknüpfungen deklarieren. Durch Einfügen von Tabelleneinträgen können nun einfache Verknüpfungen zwischen den nebenläufigen Objekten hergestellt werden, wobei die Schnittstellen der Objekte automatisch generiert werden können. Nachdem das Top-Level eines Designs, beispielsweise eines Hardware-Designs, fertiggestellt ist, können die Teilkomponenten weiter verfeinert werden. Dazu kann eine Tabelle, die bereits alle Schnittstellensignale enthält, automatisch bereit gestellt werden.

Die Einführung einer Zwischenformattabelle und die erfindungsgemäßen Verfahren zur Umsetzung ermöglichen in deutlich vereinfachter Weise die Umwandlung eines Quellprogrammcodes in einem Zielprogrammcode, verringern die Zahlen notwendiger Umsetzungsprogramme und geben in Form der Zwischenformattabelle eine übersichtliche Kontrollmöglichkeit über den Umsetzungsvorgang. Es ist sogar möglich, nach Umsetzung eines Quellprogrammcodes einer bestimmten Programmiersprache, beispielsweise VHDL, unmittelbar an der Zwischenformattabelle Modifikationen des gewünschten Designs vorzunehmen, um diese wieder in die ursprüngliche Programmiersprache, beispielsweise VHDL, zurück zu übersetzen und dadurch ohne Manipulationen am Ausgangsprogrammcode ein verändertes Design zu erhalten. Hierdurch kann also ein Editieren des Programms erreicht werden, ohne notwendigerweise Detailkenntnisse der jeweils verwendeten Programmiersprache zu haben. Gegenüber einer Notation in einer Hardware-Beschreibungssprache sind die einzelnen Informationen zu einem Design nicht über weite Teile des Code

verstreut, sondern liegen in der Zwischenformattabelle in einer übersichtlichen und kompakten Form vor.

- Als Erweiterung der vorgeschlagenen Zwischenformattabelle
- 5 könnten auch VHDL Generate Statements unterstützt werden, indem diese als spezielle Attributfelder den entsprechenden Reihen erster Dimension zugeordnet werden. Bei Nicht-Hardware-Beschreibungssprachen ergibt sich ebenfalls ein weites Benutzungsfeld, bei dem variable Konstanten und Unterprogramme umgesetzt werden können. Damit wird eine Partizipierung einer sequentiellen Beschreibung in nebenläufige Prozesse erleichtert, wie in Echtzeitumgebungen, Multitasking-Betriebssystemen und bei "multithreaded" graphischen Benutzeroberflächen vorkommen. Zudem können Daten, die nur geschrieben und niemals gelesen werden, leicht identifiziert
- 15 und gegebenenfalls aus der Datenstruktur entfernt werden. Insgesamt erleichtern die vorgeschlagenen Verfahren und die Zwischenformattabelle entscheidend die Umsetzung von Programmcodes zwischen verschiedenen Programmiersprachen sowie
- 20 die Kontrolle und Nachbearbeitung des Umsetzungsvorgangs sowie weitere Modifikationsmöglichkeiten während des Umsetzungsvorgangs.